CpE 645 Image Processing and Computer Vision

Prof. Hong Man

Department of Electrical and Computer Engineering
Stevens Institute of Technology

STEVENS Institute of Technology

Visual Information Environment Laboratory

Image Analysis

- Common image analysis operators
  - Feature detection
  - Feature extraction, segmentation
  - Feature representation
  - Feature classification
- Image analysis applications
  - medical imaging
  - security/surveillance
  - manufacturing
  - military
  - …

Feature Detection

- Feature of detection: isolated points, lines, edges
- Filtering based feature detection
  - Filter the image with a feature detection mask

$$R[n_1, n_2] = \text{image} ** \text{mask}$$

  - Extract feature based on thresholding

$$\left| R[n_1, n_2] \right| \geq T$$

- Feature detection masks generally <u>resemble</u> the feature it can detect. The filtering is essentially searching for features having strong <u>correlation</u> with the mask.

Point Detection

- Point detection: to locate isolated abnormal points
- Example: a 1-D point detection mask with length of 3

$$m=[-1, 8, -1]$$

input: 2 3 2 4 3 10 2 3 3 4 3 2

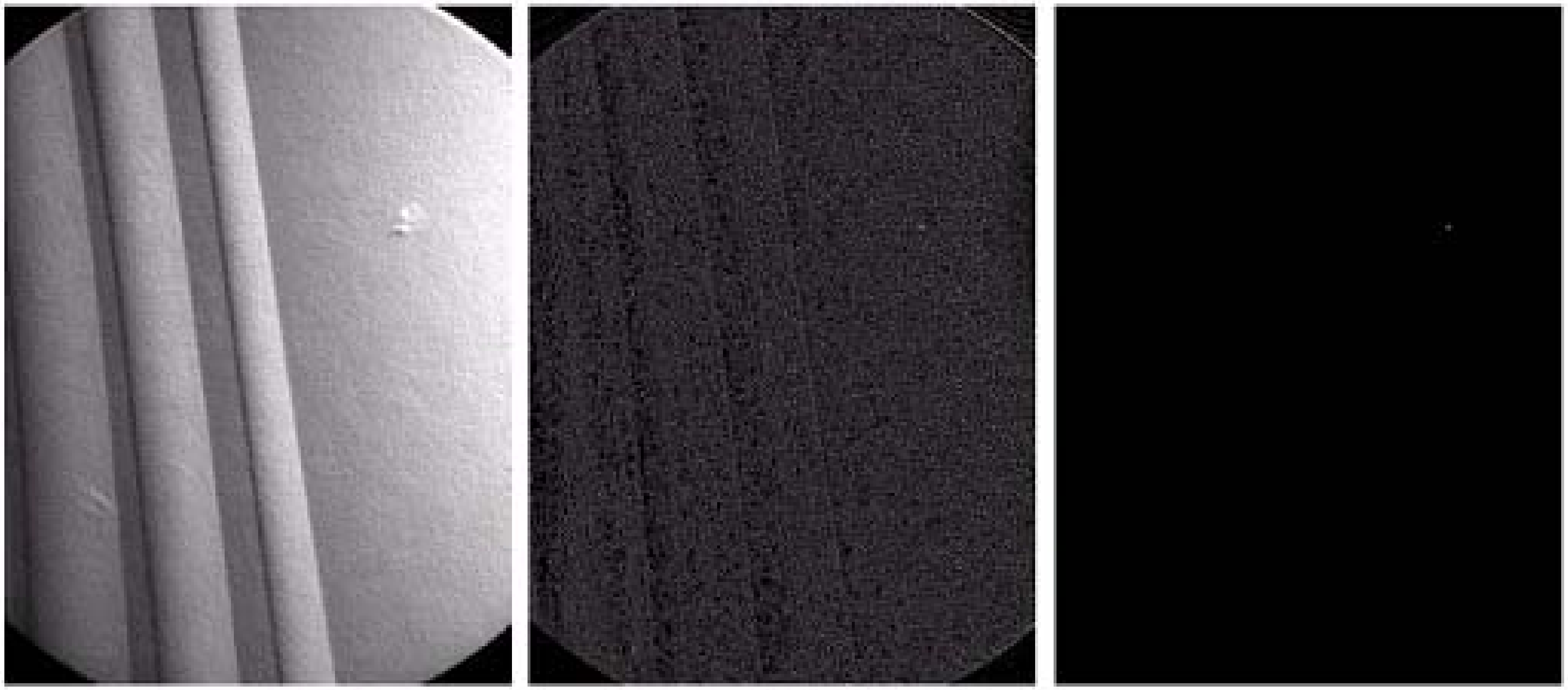
isolated point

output: 20 9 27 10 75 3 19 17 26 18 …

- A 2-D point detection mask

−1	−1	−1
−1	8	−1
−1	−1	−1

Point Detection



Line Detection

- Line detection: to detect thin lines
- Some line detection masks

−1	−1	−1
2	2	2
−1	−1	−1

Horizontal

−1	−1	2
−1	2	−1
2	−1	−1

+45°

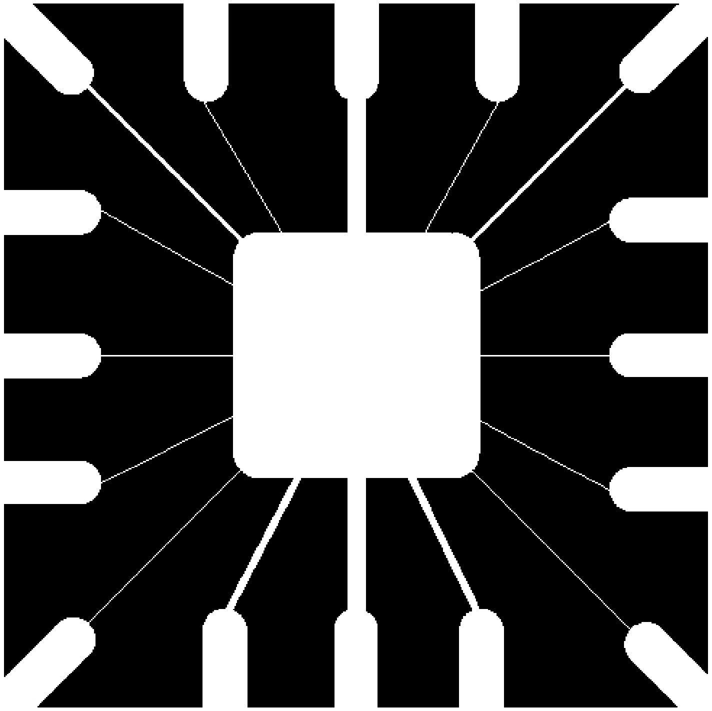
−1	2	−1
−1	2	−1
−1	2	−1

Vertical

2	−1	−1
−1	2	−1
−1	−1	2

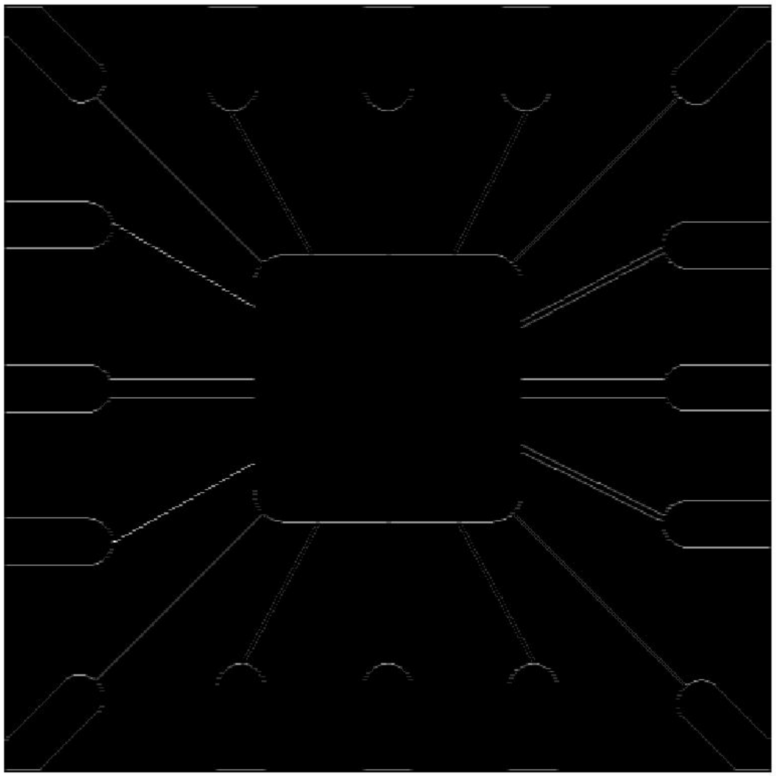
−45°

Line Detection

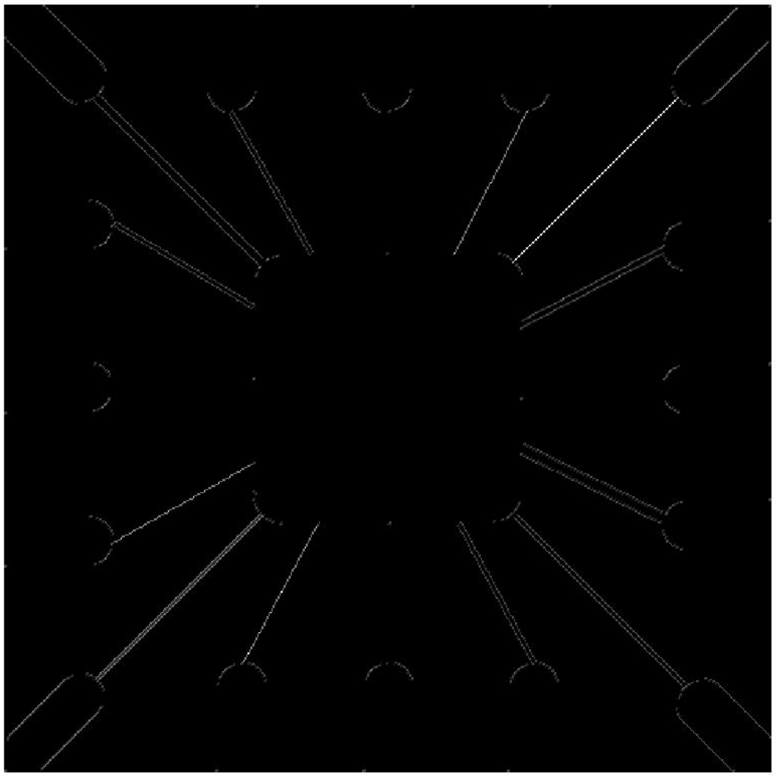


Original

Line Detection

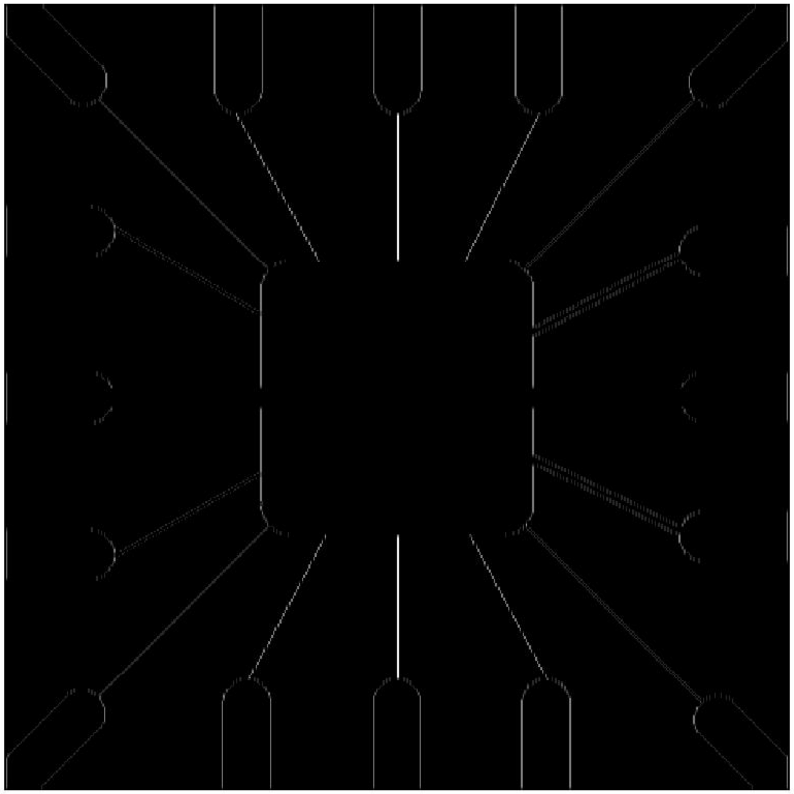


horizontal

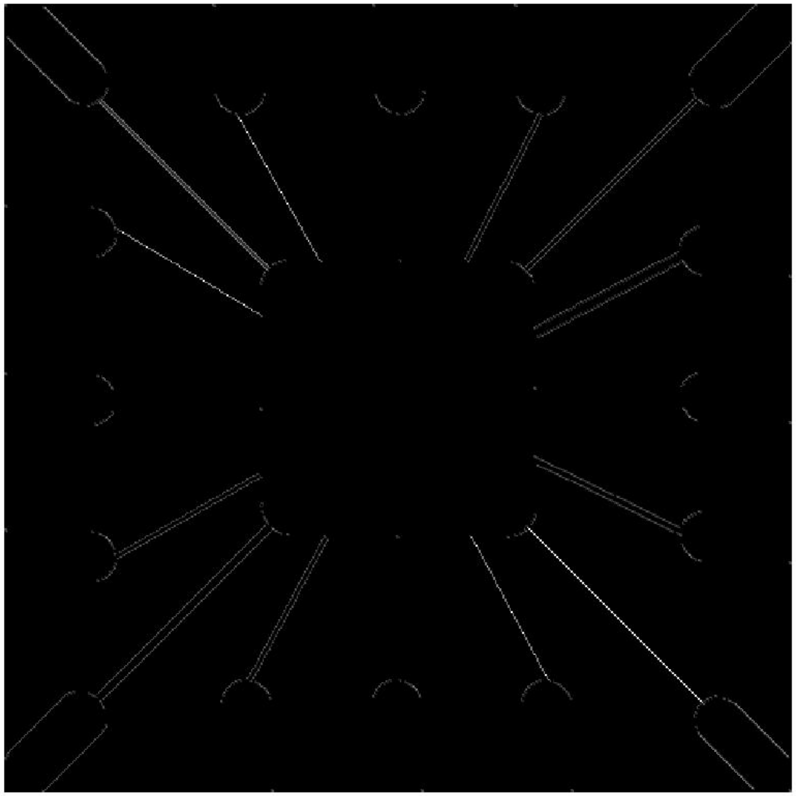


diagonal +45°

Line Detection

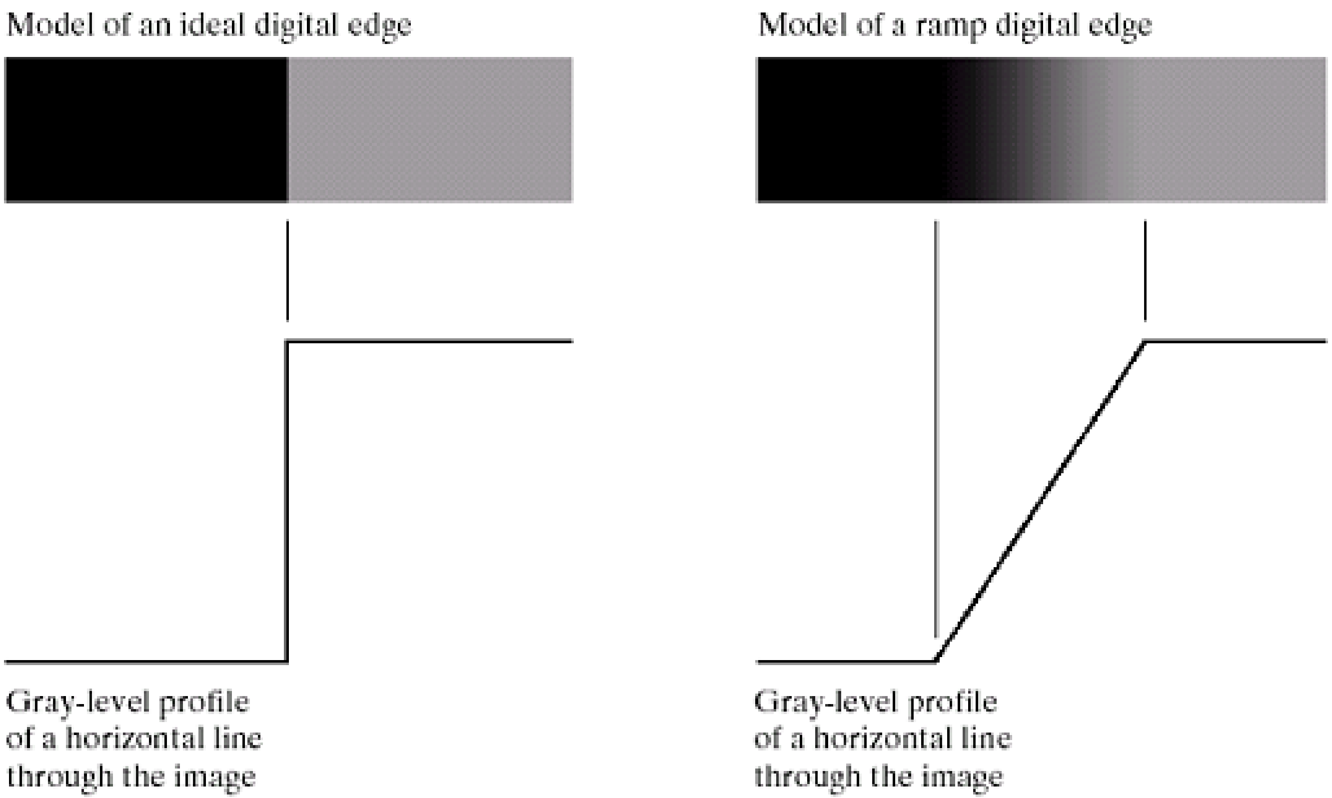


vertical

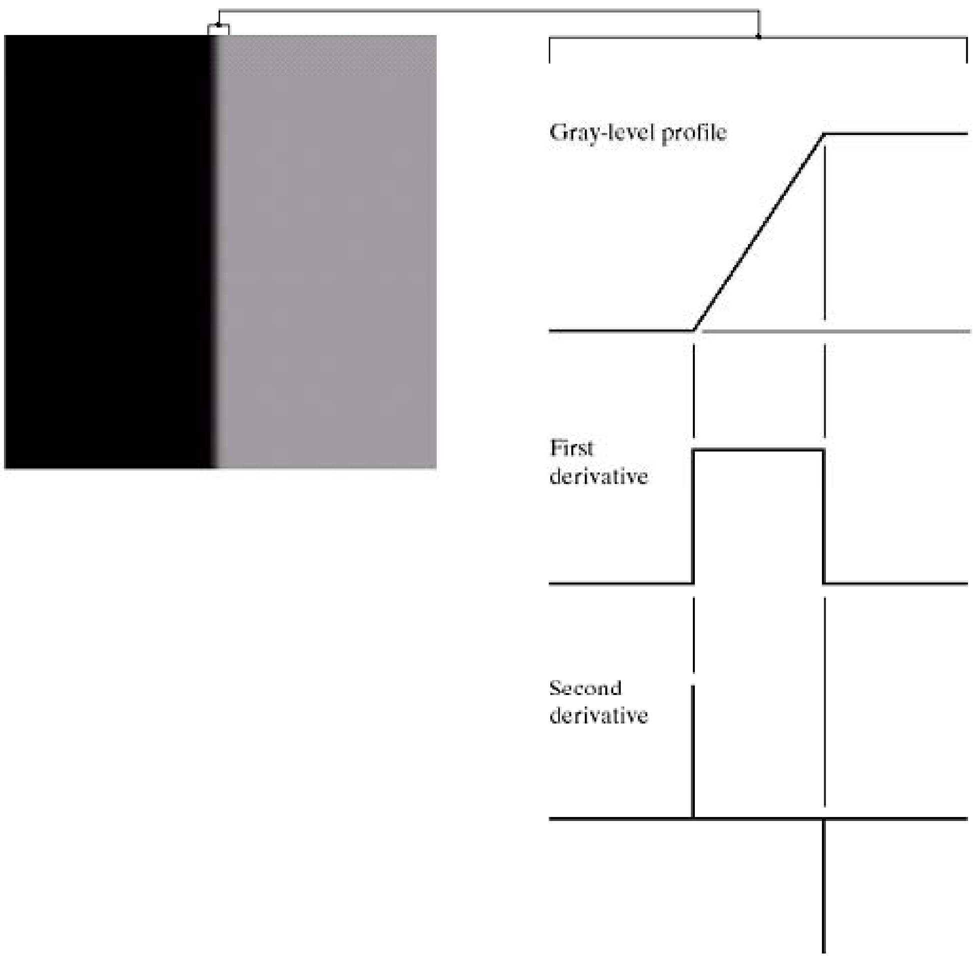


diagonal -45°

Edge Detection



Edge Detection



Edge Detection

- The gradient magnitude indicates the amount of amplitude change in neighboring pixels.
- The gradient is the 2-D equivalent of the first derivative and is defined as the vector:

$$\nabla[f(x,y)] = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

- The magnitude and direction of the gradient is

$$\nabla f = mag(\nabla[f(x,y)]) = \sqrt{G_x^2 + G_y^2}$$

$$\alpha(x,y) = \tan^{-1}\left(\frac{G_y}{G_x}\right)$$

Edge Detection

- Practical gradient calculation:
  - For a 3x3 image block

    an example of gradient

    $G_x = (z_3 + z_6 + z_9) - (z_1 + z_4 + z_7)$

    $G_y = (z_7 + z_8 + z_9) - (z_1 + z_2 + z_3)$

    and the gradient magnitude is

    $mag(\nabla f) \approx G_x + G_y$

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

- Edges are located by thresholding the magnitude of the gradient, i.e. $|mag(\nabla f)| \geq T$

Edge Detection

- Common edge detection mask pairs
- Procedure:
  - Filter image with the first mask to obtains G_x
  - Filter image with the second mask to obtain G_y
  - Calculate $mag(\nabla f)$
  - Threshold the gradient magnitude image with T.

−1	0		0	−1
0	1		1	0

Roberts

−1	−1	−1		−1	0	1
0	0	0		−1	0	1
1	1	1		−1	0	1

Prewitt

−1	−2	−1		−1	0	1
0	0	0		−2	0	2
1	2	1		−1	0	1

Sobel

Edge Detection

0	1	1
−1	0	1
−1	−1	0

−1	−1	0
−1	0	1
0	1	1

Prewitt

0	1	2
−1	0	1
−2	−1	0

−2	−1	0
−1	0	1
0	1	2

Sobel

Prewitt and Sobel masks for detecting diagonal edges.

Edge Detection

original

G_x

G_y

$mag(\nabla f)$

Edge Detection

- The Laplacian operator (2-D equivalent of the second derivative) is given by:

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

- A practical example of 1-D Laplacian operator

$$\begin{aligned}
\frac{\partial^2 f}{\partial x^2} &= \frac{\partial G_x}{\partial x} \\
&= \frac{\partial (f[i,j] - f[i,j-1])}{\partial x} \\
&= \left(f[i,j+1] - f[i,j]\right) - \left(f[i,j] - f[i,j-1]\right) \\
&= f[i,j+1] - 2f[i,j] + f[i,j-1]
\end{aligned}$$

Edge Detection

- Two typical Laplacian operators
- Laplacian operators may produce double edges, and are sensitive to noise.

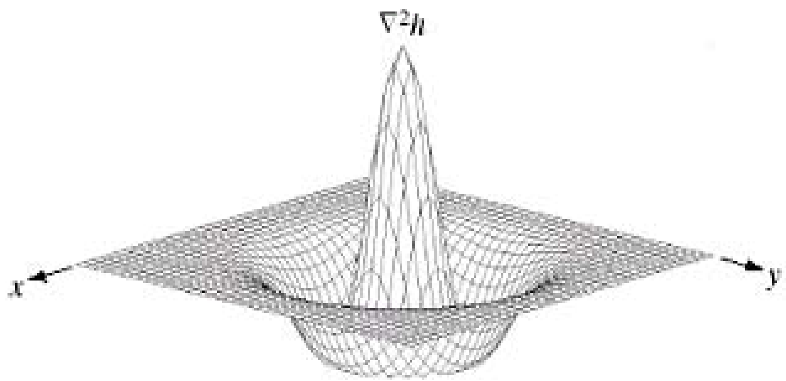
0	−1	0
−1	4	−1
0	−1	0

−1	−1	−1
−1	8	−1
−1	−1	−1

- To apply Laplacian operator, the image is usually first smoothed by a Gausian filter.
- The combined effect of Gausian smooth filter and Laplacian edge detector is referred as the **Laplacian of a Gaussian (LoG).**

Edge Detection

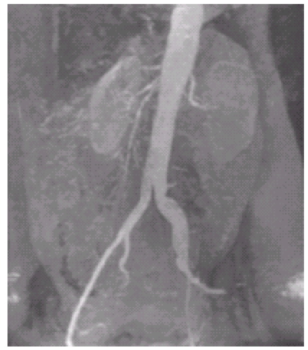
- An example of LoG operator.
- Procedure:
  - Filter the image with the LoG operator.
  - Threshold the resulting image
  - Locate the zero-crossing points in the thresholded image, which represent the edge information.



0	0	−1	0	0
0	−1	−2	−1	0
−1	−2	16	−2	−1
0	−1	−2	−1	0
0	0	−1	0	0

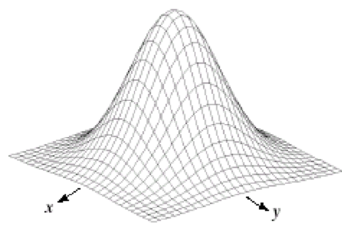
Edge Detection

original



Sobel gradient

Gaussian

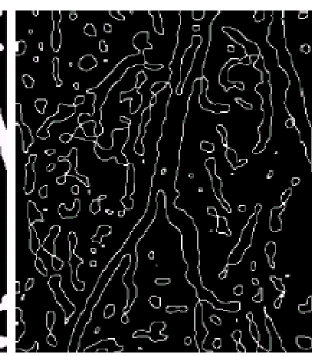


−1	−1	−1
−1	8	−1
−1	−1	−1

Laplacian

LoG
result

Threshold



Zero-Crossing
result

Edge Linking

- Edges detection may yield disconnected edges.
- Local Processing: analyze the characteristics of pixels in a small neighborhood about every point in an image that has undergone edge detection. All points that are similar are linked, forming a boundary of pixels that share some common properties:
  - Strength of the response of the gradient operator

$$\left|\nabla f(x, y) - \nabla f(x_0, y_0)\right| \le T$$

  - Direction (angle) of the gradient

$$\left|\alpha(x, y) - \alpha(x_0, y_0)\right| \le A$$

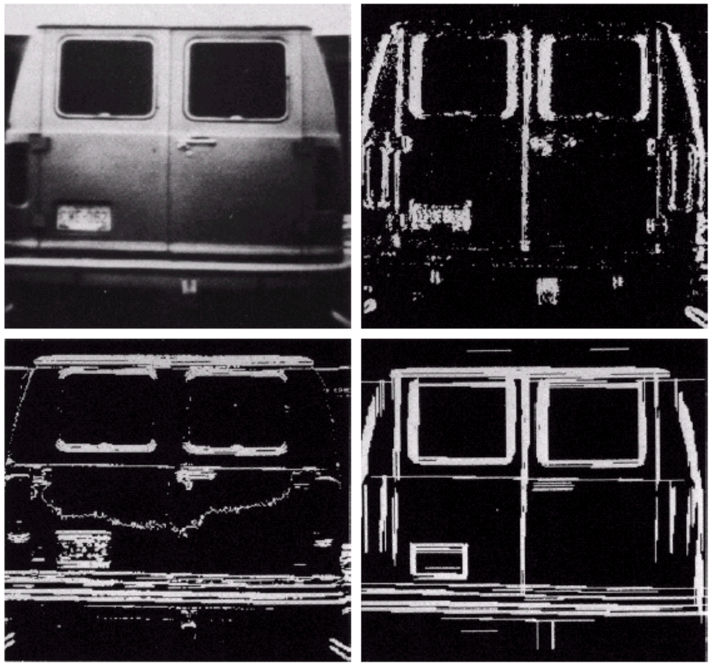
  where both T and A are non-negative similarity thresholds

Edge Linking

- Example: criteria for linking:
  - gradient magnitude greater than 25 AND
  - gradient directions did not differ by more than 15°

a b
c d

FIGURE 10.16
(a) Input image.
(b) G_y component of the gradient.
(c) G_x component of the gradient.
(d) Result of edge linking. (Courtesy of Perceptics Corporation.)



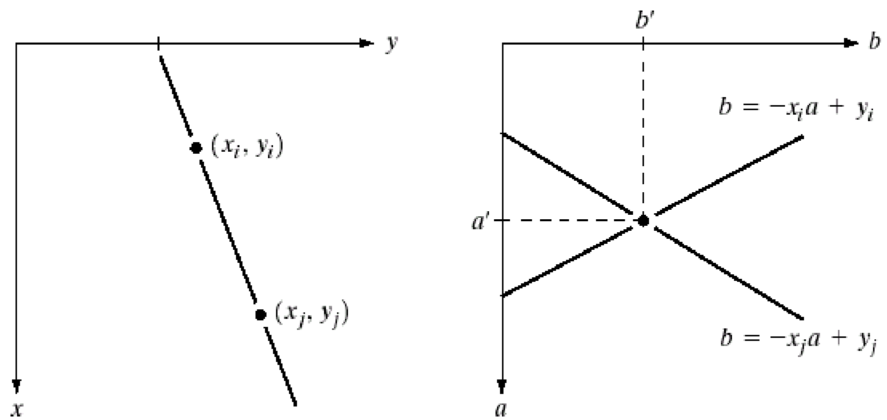
Edge Linking

- Global processing uses global relationship between pixels to link them by determining whether they lie on a curve of specified shape.
- Simple case: straight lines
  - Suppose for $\boldsymbol{n}$ points in an image, we want to find subsets of these points that lie on straight lines.
  - One solution: find all lines determined by every pair of points and then find all subsets of points that are close to particular lines.
  - This solution involves finding $\boldsymbol{n}^2$ lines and performing $\boldsymbol{n}^3$ comparisons of every point to all lines – computationally intensive

Edge Linking

- Hough (1962) proposed an alternative approach commonly referred to as Hough transform.
- Consider a point (x_i, y_i), and all lines pass (x_i, y_i) satisfy the equation: $$y_i = ax_i + b$$
  we can rewriting this equation as:
  $$b = -x_i a + y_i$$
- Considering the ***ab***-plane (parameter space) yields the equation of a single line for a fixed pair of (x_i, y_i)
- The computational attractiveness of the Hough transform arises from subdivision of the parameter space into so-called accumulator cells

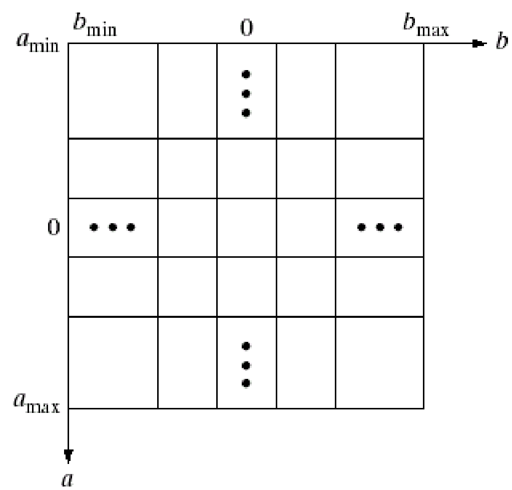
Edge Linking



a b

FIGURE 10.17
(a) xy-plane.
(b) Parameter space.

FIGURE 10.18
Subdivision of the parameter plane for use in the Hough transform.

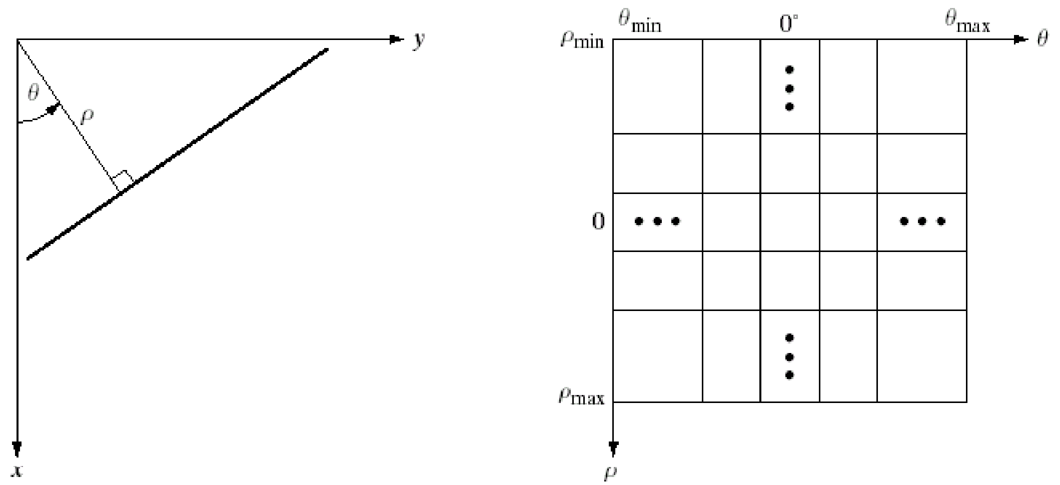


Edge Linking

- The Hough transform procedure:
  - Initialize all accumulator cells with value of zero
  - For every point (x_k, y_k) in the image plane, we let the parameter a equal each of the allowed subdivision values and solve for the corresponding b. The resulting b is rounded off to the nearest allowed value in the b axis. (Note: a is slope and b is intercept value)
  - If a choice a_p results in solution b_q, we let accumulator value $A(p, q)=A(p, q)+1$
  - At the end of the procedure, a value M in $A(i, j)$ corresponds to M points in the (x, y) plane lying on the line $y=a_i x+b_j$

Edge Linking

- A problem with using the equation $y=ax+b$ to represent a line is that both the slope and intercept approach infinity as the line approaches the vertical.
- One way around this difficulty is to use the normal representation of a line: $x\cos\theta + y\sin\theta = \rho$



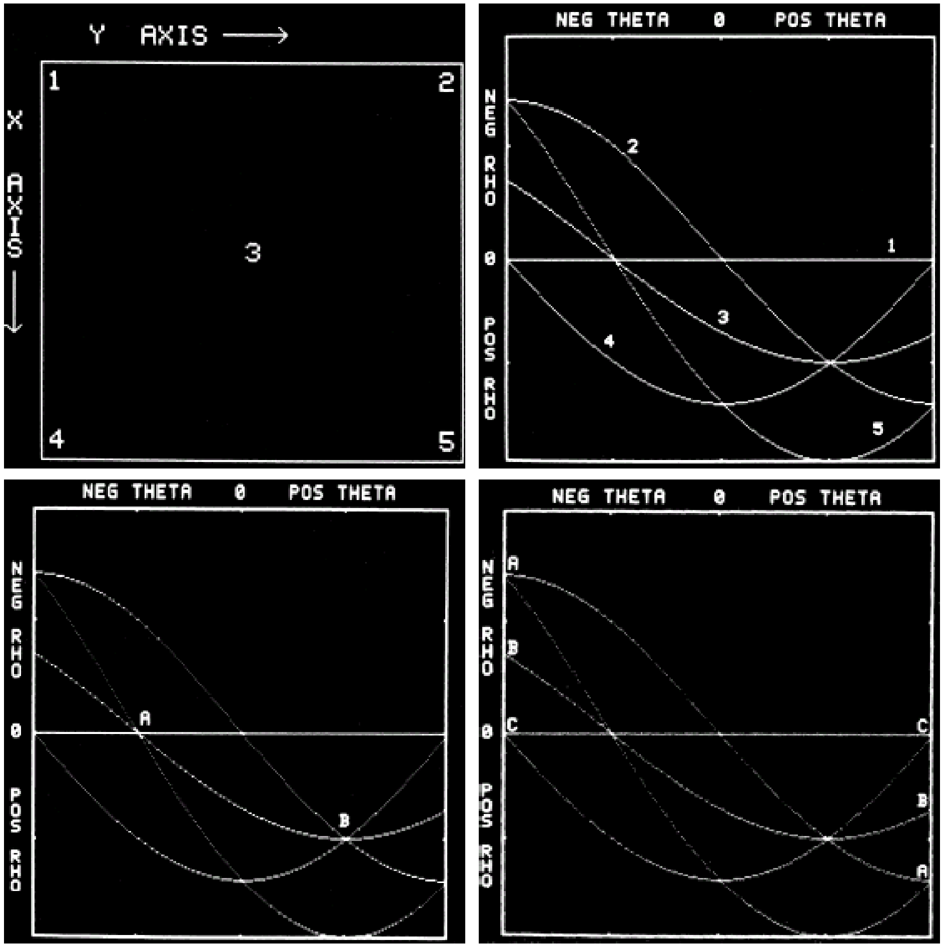
a b

FIGURE 10.19
(a) Normal representation of a line.
(b) Subdivision of the $\rho\theta$-plane into cells.

Edge Linking

a b
c d

FIGURE 10.20 Illustration of the Hough transform. (Courtesy of Mr. D. R. Cate, Texas Instruments, Inc.)



Edge Linking

- The range of θ is $\pm 90^o$ measured w.r.t. the x-axis. The range of ρ is $\pm\infty$
- In previous figure (figure10.20),
  - (a) 5 isolated points in xy-plane.
  - (b) each point is mapped to one sinusoidal curve in the $\theta\rho$-plane.
  - (c) colinearity detection property of Hough transform: point A denotes the intersection of the curves for point 1, 3 and 5, the location of A indicates these three points lie on a straight line passing through the origin ($\rho = \mathbf{0}$) and oriented at -45^o; point B indicates that points 2, 3 and 4 lie on a straight line oriented at 45^o and away from the origin by half of the diagonal distance.

Edge Linking

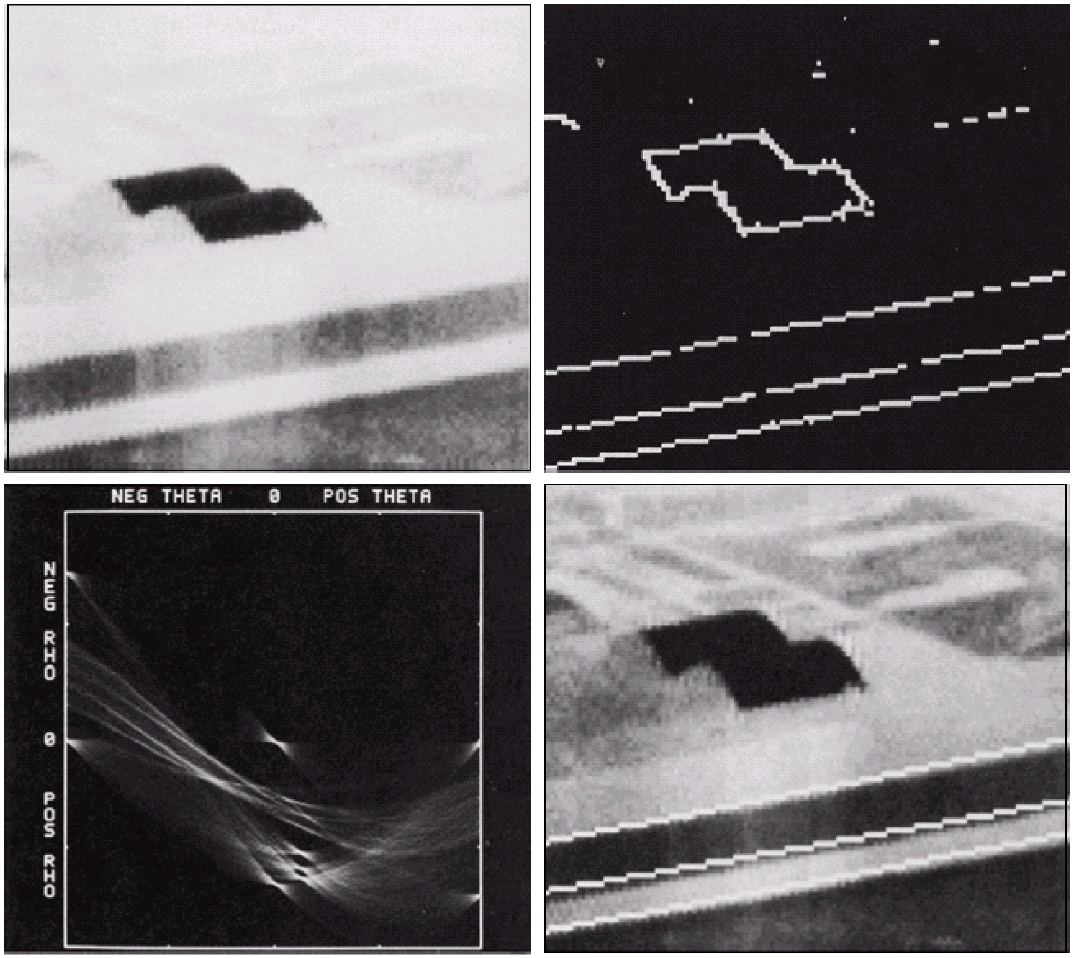
- (d) Hough transform exhibits a reflective adjacency relationship at the right and left edges of the parameter space. Points A, B and C indicate such manner, in which θ and ρ change sign at the $\pm 90^{o}$ boundaries.

- Although the focus so far has been on straight lines, the Hough transform is applicable to any function of the form $\boldsymbol{g(V, C)=0}$ where $\boldsymbol{V}$ is a vector of coordinates and $\boldsymbol{C}$ is a vector of coefficients.
- Example: points lying on a circle $\boldsymbol{(x-c_1)^2+(y-c_2)^2=c_3^2}$ can be detected using the same approach on the $\boldsymbol{(c_1, c_2, c_3)}$ plane

Edge Linking

- An edge linking approach based on Hough transform:
  - Compute the gradient of the image and threshold it to obtain a binary image
  - Specify subdivisions in the $\rho\theta$–plane
  - Examine the counts of the accumulator cells for high pixel concentrations
  - Examine the relation (mostly for continuity) between the pixels in a chosen cell. The continuity is measured by the distance between disconnected pixels identified during traversal of the set of pixels corresponding to a given accumulator cell.

Edge Linking



a b
c d

FIGURE 10.21 (a) Infrared image. (b) Thresholded gradient image. (c) Hough transform. (d) Linked pixels. (Courtesy of Mr. D. R. Cate, Texas Instruments, Inc.)

Edge Linking

- The set of pixels will be linked according to the criteria:
  - (1) they belong to one of the N accumulator cells with the highest count, AND
  - (2) no gaps are larger than a threshold T, where a gap at any point is measured by the distance between that point and its closest neighbor.
- In previous figure (figure 10.21) (d), N = 3, and T = 5 pixel.

Segmentation

- Image segmentation is a method to partition an image into several regions.
- Segmentation is done based on certain homogeneity criteria, such as gray-level intensity, motion, depth, semantic meaning, etc…
- Segmentation is an important image analysis stage to extract object shape information for further characterization and classification.
- Segmentation based on intensity value is usually called thresholding.

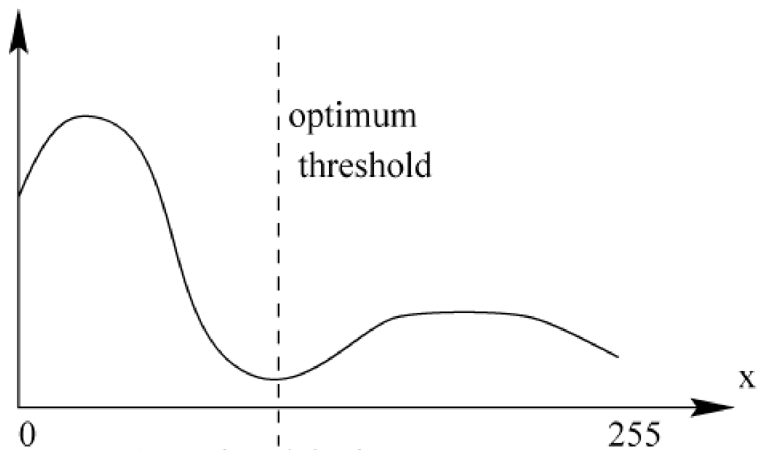
Segmentation Examples

Thresholding

Edge detection

Thresholding

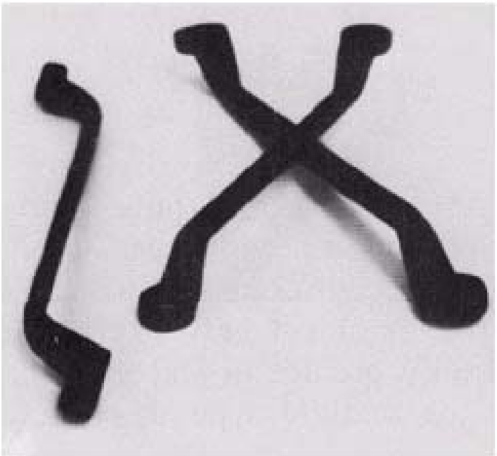
- The simplest thresholding is global thresholding, which applies a threshold T to all pixels in the image.
- It is necessary that the objects and background have sufficient contrast.
- The threshold value should be determined based on the histogram.



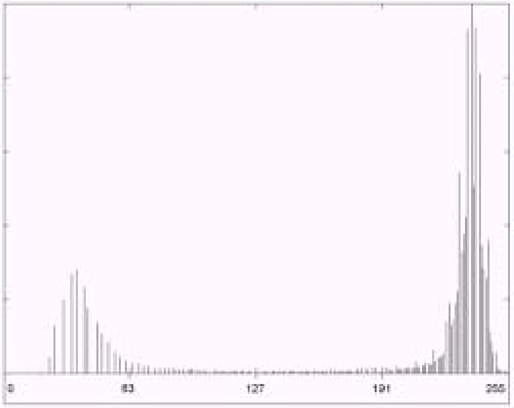
(a) Bimodal Histogram

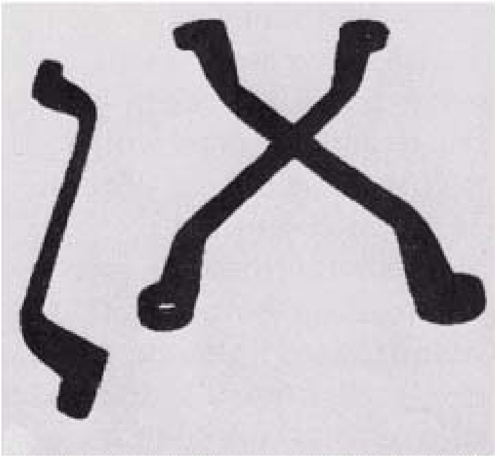
Thresholding

Original



Histogram





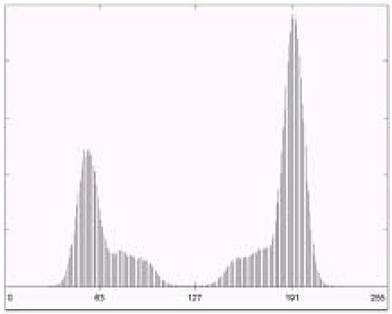
Thresholded
at 127

Thresholding

- Iterative global threshold estimate algorithm
  1. Select an arbitrary initial threshold T
  2. Segment the image to obtain two pixel groups: group 1 pixel values > T, group 2 pixel values ≤ T.
  3. Compute the average values μ_1 and μ_2 of group 1 and group 2 pixels
  4. Update the threshold value $T = \frac{1}{2} (\mu_1 + \mu_2)$.
  5. Repeat step 2 to 4 until the changes in T become very small (smaller than a preset number).

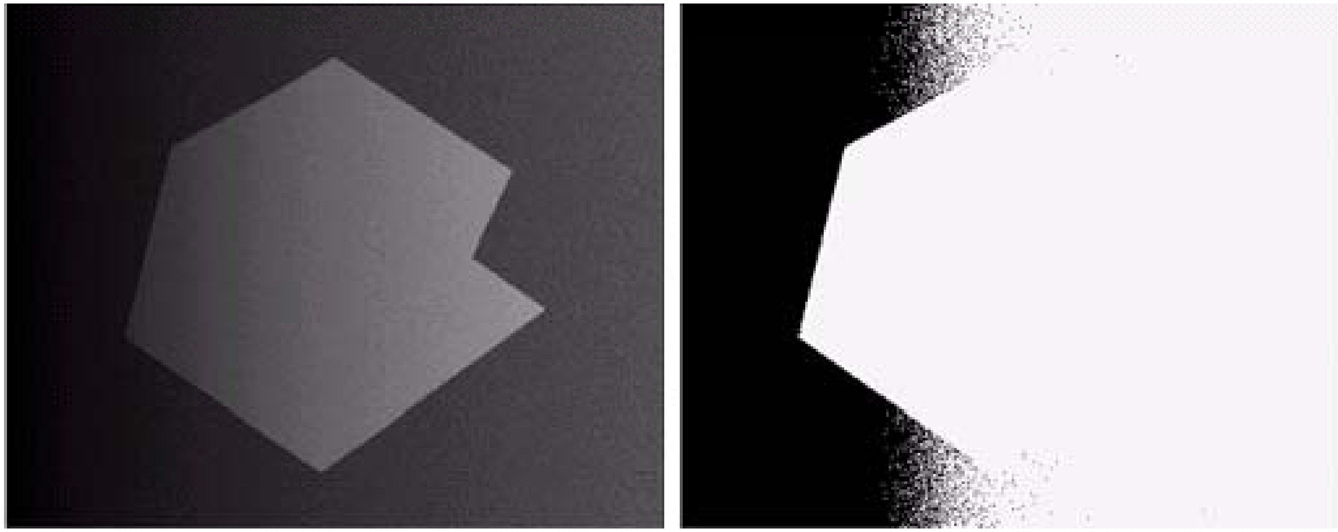
Thresholding

Thresholding through iterative threshold estimation



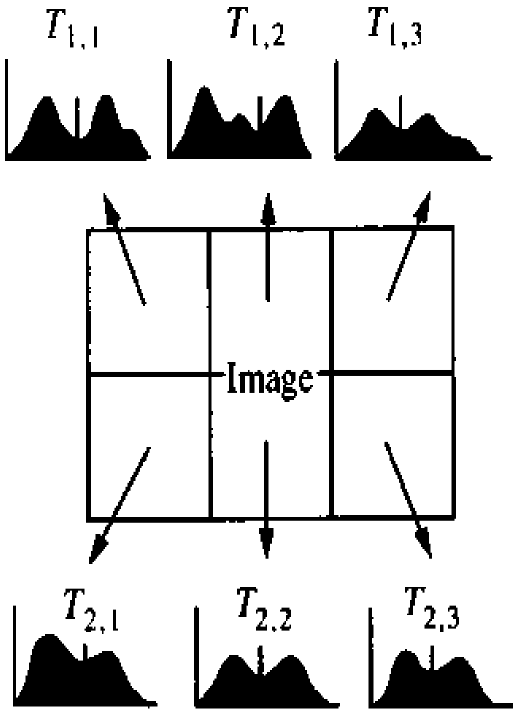
Thresholding

- Simple global thresholding is very sensitive to local illumination condition



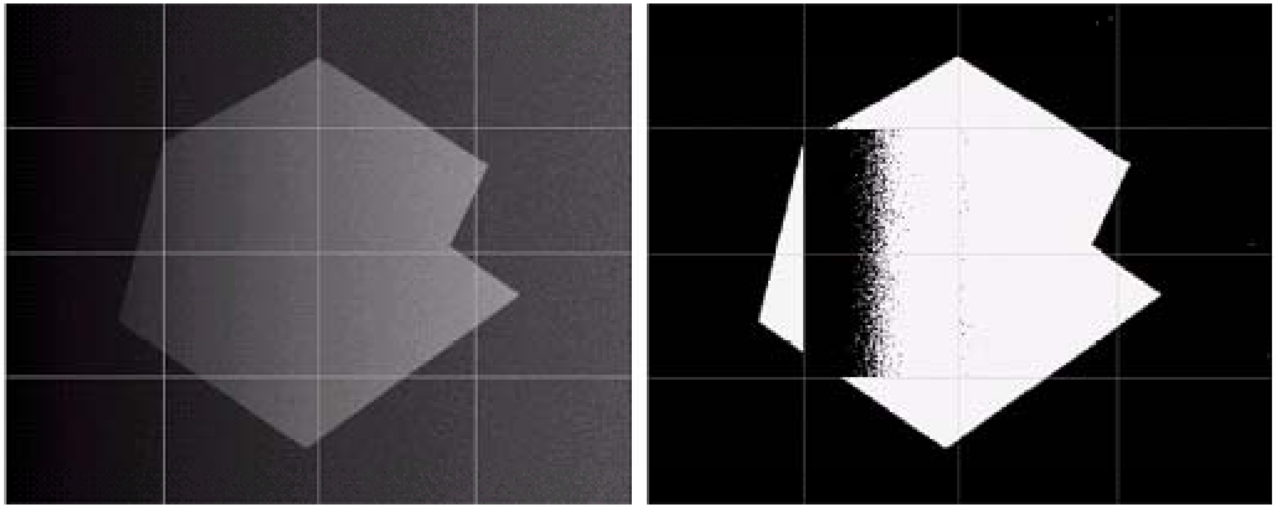
Thresholding

- This situation can be addressed by using adaptive thresholding rather than global thresholding.
- A basic adaptive thresholding
  – partition an image into subimages
  – calculate the best threshold for each of these subimages.
- The hope is, the contrast of the object could be higher within each subimage, i.e. the local histograms are bimodal.



Thresholding

- Example of adaptive thresholding.



Thresholding

- In the presence of noise, global and even adaptive thresholding methods may result in spurious pixels.
- The problem is that a noise spike in the image may have a high amplitude value that cannot be distinguished directly from an object value.
- Two-level thresholding:
  - Select a high level threshold. Capture the main pixels in all the objects
  - Lower the threshold. Pixels are added to the list of object pixels if their amplitudes are above the threshold AND these pixels are <u>connected</u> to an object pixel.

Thresholding

- Two level thresholding

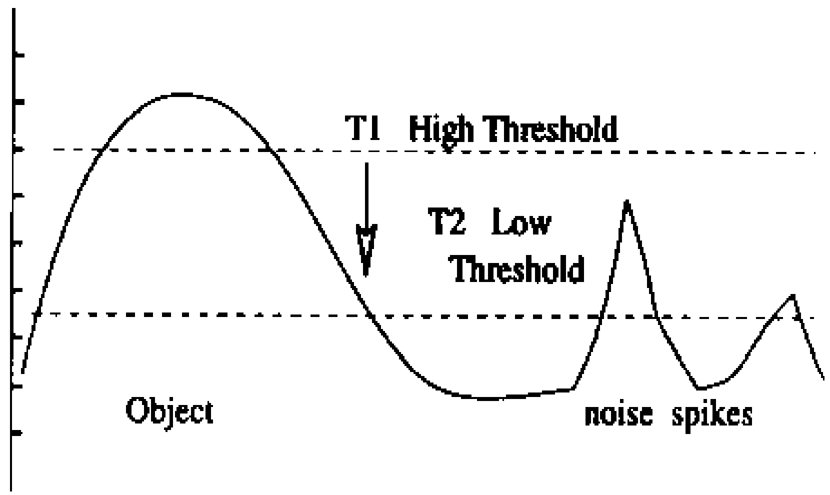
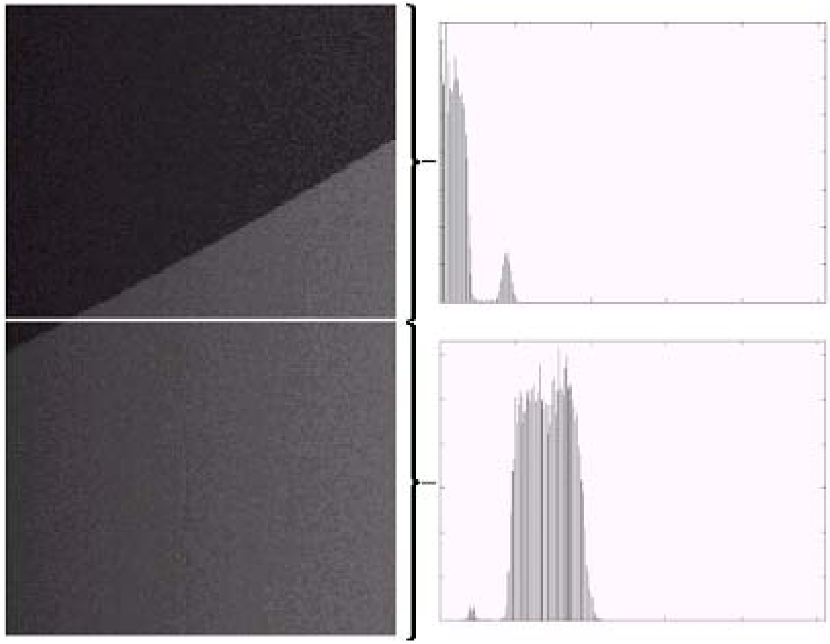


Figure 10.2: Illustration of the two-level threshold segmentation method.

Thresholding

- Histogram distributions generally depend on the size of objects and background



Thresholding

- We can select only those pixels that are on or near the edge between object and background in calculating the histogram
- The resulting histogram should have similar heights for object and background, and have valley around the middle value.
- Indication of whether a pixel is on an edge may be obtained by computing its gradient and Laplacian.
- The test

$$s(x, y) = \begin{cases} 0 & \text{if } \nabla f < T \\ + & \text{if } \nabla f \geq T \quad \text{and } \nabla^2 f \geq 0 \\ - & \text{if } \nabla f \geq T \quad \text{and } \nabla^2 f < 0 \end{cases}$$

Thresholding

- $\nabla f < T \Rightarrow$ not an edge

  $\nabla f \geq T$ and $\nabla^2 f \geq 0 \Rightarrow$ dark side of an edge

  $\nabla f \geq T$ and $\nabla^2 f < 0 \Rightarrow$ light side of an edge
- These edge pixels can be used to obtain local thresholds (note, gradient and Laplacian are local operators).

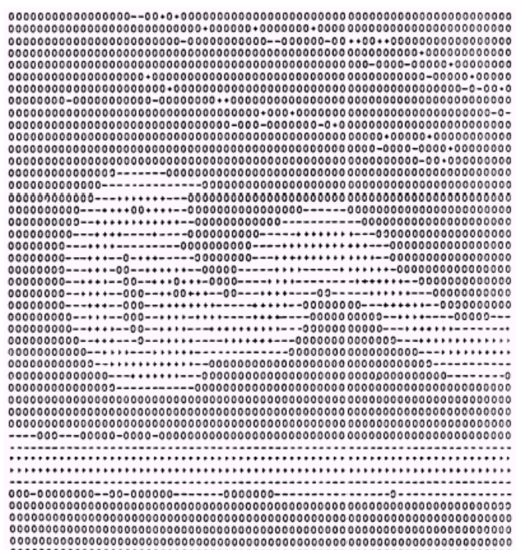


FIGURE 10.36 Image of a handwritten stroke coded by using Eq. (10.3-16). (Courtesy of IBM Corporation.)

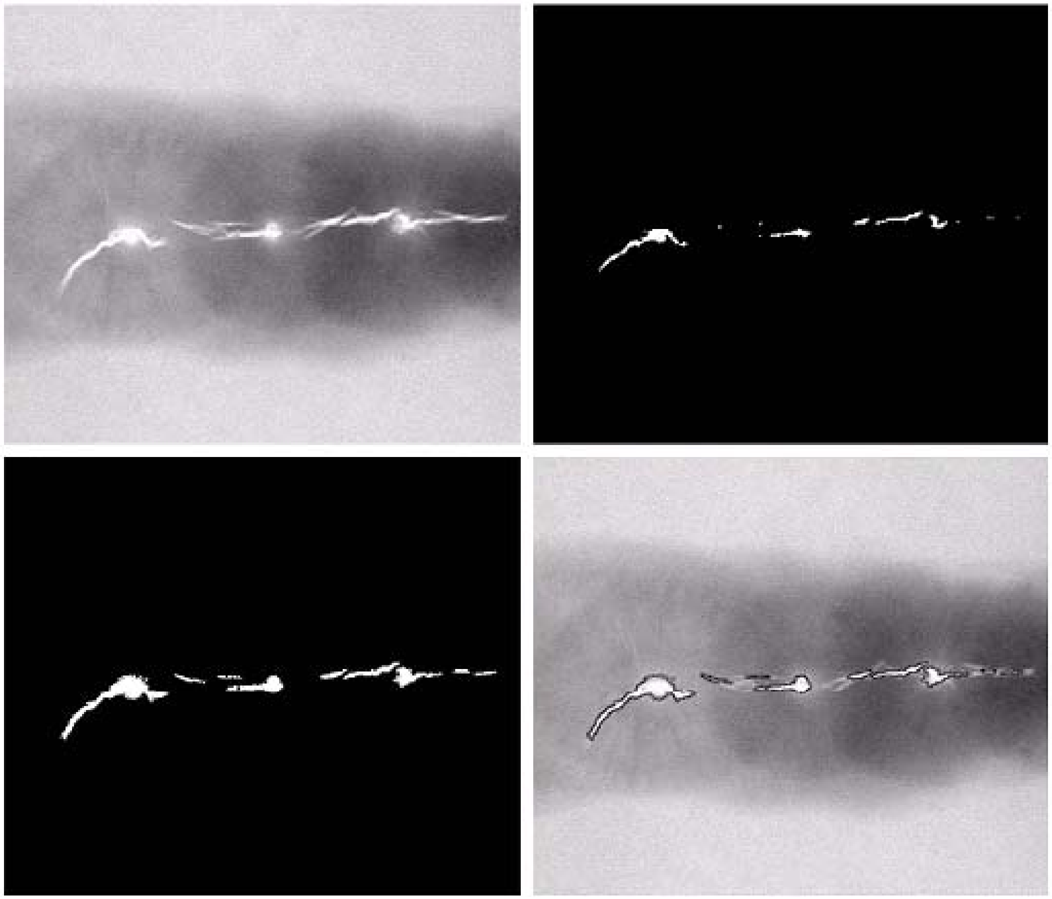
Region Based Segmentation

- Let R represent the entire image region. Region based segmentation is a process that partitions R into **n** subregions, $R_1, R_2, \ldots, R_n$ such that
  - R_i is a connected region
  - intersection between regions is null
  - all pixels in R_i share the same property P
  - pixles in R_i and R_j do not share this property P

Region Based Segmentation

- **Region grow** is a procedure that groups pixels or sub-regions into larger regions.
  - Specify a set of "seed" points
  - Append a neighboring pixel to these seed points if
    - the neighboring pixel is connected to this seed point
    - the neighboring pixel satisfies the predetermined properties (intensity, color, texture etc.) for this seed point.
  - the newly grown pixels become new seed points and repeat the process until no new pixel can be found.

Region Based Segmentation



original

seed points

grown region

extracted boundary

Region Based Segmentation

- Fundamental difficulties in region growing
  - Selection of initial seeds that properly represent regions of interest
  - Selection of suitable properties for including points in the various regions during the growing process
- Selection of starting points often can be based on the nature of the problem, i.e., in infrared images, choose the brightest pixel as the seed point

Region Based Segmentation

- Example criteria can be used in region growing

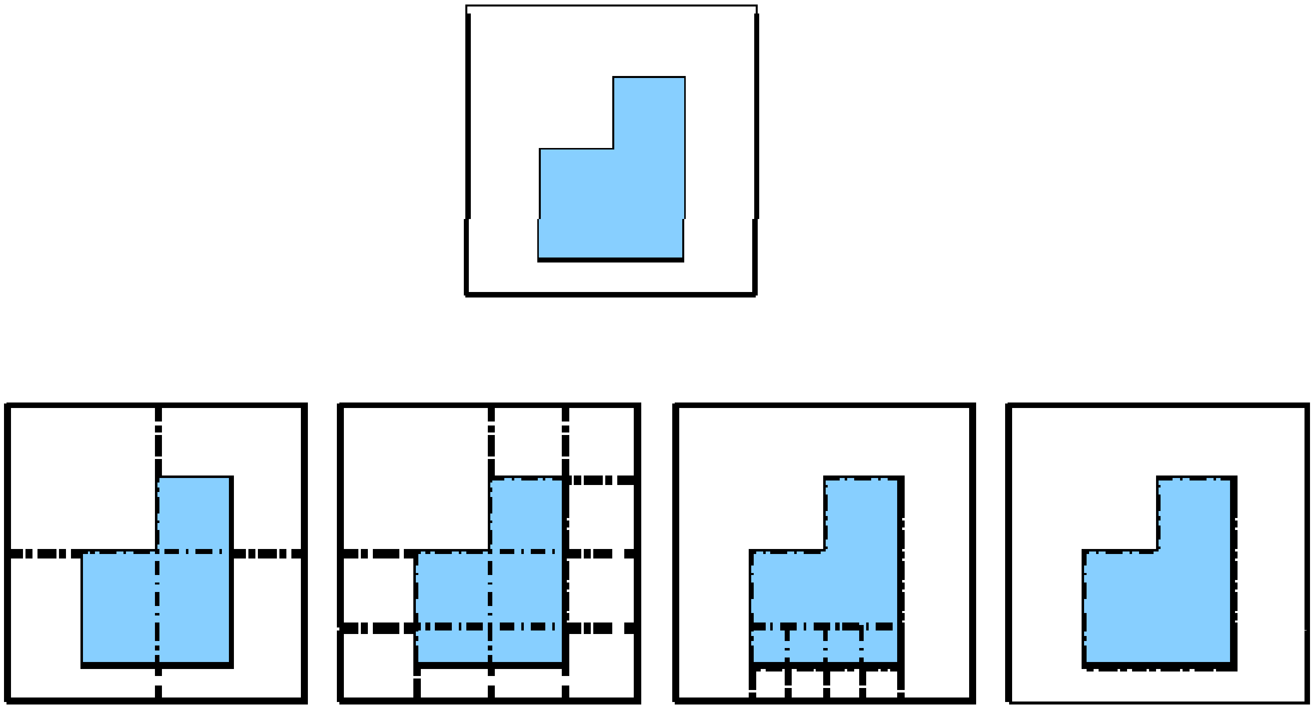
  (1) the absolute difference between the seed and a candidate point not exceed 10% of the dynamic range of the image, and

  (2) that any pixel added to the region be 8-connected to at least one pixel previously included in the region.
- Additional criteria that increase the power of region-growing algorithm utilize the concept of size, shape of region being grown, etc…
- Region growing may be a useful method for selecting a few regions in an image, but it is rarely the method of choice for complex images containing many regions

Region Based Segmentation

- **Region splitting and merging**: an approach to subdivide the image into a set of arbitrary regions and then merge and/or split based on the logical predicate $\boldsymbol{P(R)}$ – the predefined properties.
- Procedure
  - split any region $\boldsymbol{R_i}$ into disjointed subregions if this region contains pixel that does not satisfy the $\boldsymbol{P(R_i)}$
  - merge adjacent regions $\boldsymbol{R_j}$ and $\boldsymbol{R_k}$ if all their pixels satisfy a common $\boldsymbol{P(R_j \cup R_k)}$
  - stop when no further merging or splitting is possible

Region Based Segmentation



Region Based Segmentation

- Example criteria can be used in region growing
  $P(R_i)=TRUE$ if at least 80% of the pixels in R_i have the property

  $$\left| z_j - m_i \right| \le 2\sigma_i$$

  where

  z_j: gray level of the j^{th} pixel in R_i,

  m_i: mean gray level of the region R_i

  σ_i: standard of deviation of the gray levels in R_i